Algorithms, Section 2
Spring 2003
Homework 10b
Due Tuesday, April 15

The 2nd midterm will be another take-home, available Thursday, April 17th, after class, and due back Monday night, April 21st, at midnight. The focus of this exam will be Graph Algorithms. You will be able to use three unlimited blocks of time in finishing this exam.

1. [**10 Points**] **Trying out the Network Flow Algorithm!** CLRS, Exercise 26.2-2 (page 663). Note that the "Edmonds-Karp Algorithm" is just the algorithm we did in class (called "Ford-Fulkerson") but using breadth-first search (rather than depth-first search) to find augmenting paths. In your solution, show both the augmenting path and the residual network after each iteration as on page 659.

2. [**20 Points**] **Review of Network Flow Proofs.** In class we proved four fundamental theorems which, in the end, allowed us to show that our network flow algorithm does indeed find a maximum flow from s to t. These four theorems were:

   (a) The Cut Theorem
   (b) The Capacity Theorem
   (c) The Max-Flow Min-Cut Theorem
   (d) The "Greed is Good" Theorem for Network Flows

   Note that our proofs and approach are easier than those given in the book. Your task is to carefully write out the proofs of each of these four theorems as presented in class. Your grade will be based largely on how clearly and precisely you write your proofs.

3. [**15 Points**] **Graph Partitioning!** Given an *undirected* graph G, a *partition* of the graph is a division of the vertices into two sets A and B such that every vertex is in exactly one of A and B. The only constraint on A and B is that neither set can be empty. The *crossing number* of the partition is the number of edges with one endpoint in A and one endpoint in B. Professor Lai would like to find an algorithm which finds a partition with the smallest possible crossing number. (Fast graph partitioning algorithms are used in a variety of applications ranging from data clustering to circuit design.)

   (a) Describe your algorithm. (You may use existing algorithms to help!)
   (b) Explain clearly why your algorithm is correct.
   (c) Derive the running time of your algorithm. (Exponential time algorithms are not good!)

4. [**15 Points**] **FUHLishness!** The Harvey Mudd **F**aculty **U**nicycle **H**ockey **L**eague (FUHL) has four teams numbered 1 through 4. Each team has lost a certain number of

games so far, as shown in Table 1 below, and has a certain number of games remaining against the other teams in the league, as shown in Table 2. The league champion is the team with the minimum number of losses at the end of the season. We wish to know if Team 4 has any hope of being the league champion, in spite of the fact that it has lost so many games already.

Team number	Losses so far
1	1
2	4
3	2
4	10

Table 1

Remaining schedule

Team number			
3	4		
2	5	4	
1	7	4	2
	4	3	2
	Team number		

Table 2

(a) Show how this problem can be cast as a network flow problem by constructing the corresponding flow network.

(b) Now solve the network flow problem. For each iteration of the algorithm, show the residual graph and the flow in the network so far.

(c) Use the maximum flow to determine a scenario (which teams win and lose) in which Team 4 is the league champion.